Application of Particle Swarm Optimization to Formative E-Assessment in Project Management

Maria-Iuliana DASCĂLU
Academy of Economic Studies, Bucharest, Romania
mariaiuliana.dascalu@gmail.com

The current paper describes the application of Particle Swarm Optimization algorithm to the formative e-assessment problem in project management. The proposed approach resolves the issue of personalization, by taking into account, when selecting the item tests in an e-assessment, the following elements: the ability level of the user, the targeted difficulty of the test and the learning objectives, represented by project management concepts which have to be checked. The e-assessment tool in which the Particle Swarm Optimization algorithm is integrated is also presented. Experimental results and comparison with other algorithms used in item tests selection prove the suitability of the proposed approach to the formative e-assessment domain. The study is presented in the framework of other evolutionary and genetic algorithms applied in e-education.

Keywords: *Particle Swarm Optimization, Genetic Algorithms, Evolutionary Algorithms, Formative E-assessment, E-education*

1 Introduction

Optimization problems are issues of real interest in many domains. Because the topic of finding the best solution from all possible solutions turned out to be hard, near-optimal solutions became the chosen variant in many decision cases. Evolutionary algorithms (EAs) are preferred by researchers to find the near-optimal solution to their problems [1]. Evolutionary algorithms are based on the natural process of evolution and use the same terminology. EAs demonstrate self organization and adaptation similar to the way that the fittest biological organism survives and reproduces. The basic idea of EAs is the continuous evaluation of a set of points, which are called population. Each point, called individual, is evaluated in parallel with the other available points. The evaluation takes place using a fitness function. The continuous evaluation stops when a predefined criterion is fulfilled. An EA has embedded an iteration mechanism. A general description of this iteration mechanism would be [2]:

- A population is created with a group of individuals created randomly;
- The individuals are evaluated with a fitness function;
- A selection method is applied and some individuals are selected based on their fitness function;
- The individuals reproduce to create one or more offspring;
- The offspring are mutated randomly until a certain stop criterion is met (for example, a certain number of generations has been exceeded);

The main elements which have to be adapted in an EA are: representation of individuals (binary coding or real coding [2]), fitness function, reproduction method (how new individuals appear from existing ones), and selection criteria. The above example of EA algorithm suits best for genetic algorithms, but there are also other related techniques associated with EAs, such as: ant colony optimization, bees' algorithm, Gaussian adaption, particle swarm optimization and so on and so forth. Usually, different evolutionary techniques are combined with artificial intelligence methods, obtaining the so-called hybrid solutions: genetic algorithms and neural nets are used for adjusting parameters in fuzzy systems, genetic algorithms can improve the learning mechanism in neural networks or the selection in Item Response Theory, case-based reasoning conducts to better performances of expert systems [3]. EAs are applied in various fields, such as: routing and scheduling, robotics, medicine, learning optimizations.

The purpose of the current study is not to describe the mechanism behind a typical EA, but to present the application of EAs in the e-education domain: a detailed literature review regarding the application of EAs in e-education is provided in the second section of the study. Then the focus of the presentation is on particle swarm optimization (PSO) method: the third section describes the PSO idea, the differences between it and a classical EA and the particularization of PSO to efficiently resolve a particular problem of e-education, namely the formative e-assessment. The forth section described the e-assessment tool which benefits from the advantages of the proposed PSO-based solution. The fifth section presents the results and the validation techniques useful in checking the efficiency of the PSO-based e-assessment. The last sections contain conclusions, future directions and bibliographical resources used in elaborating the PSO solution.

2 Application of evolutionary algorithms to e-education

Before describing different EAs applied to education domain, one should understand the problem which is addressed when using these algorithms in e-learning related activities - personalizing the learning process: “personalization is essential if e-learning is to fulfill its potential in mainstream education. It has major pedagogic benefits, it provides a cost-effective means of facilitating the uptake of appropriate information from the Internet, and it has a humanizing effect upon educational experiences mediated by technology.”[4] The creators of web-based education systems are more and more preoccupied by taking into account the individual learning orientation and level [5] [6] [7] [8]. The e-learning environments are customized, so that the benefits gained at the end of the e-learning process are as large as possible. The “personalized learning” issue in e-education is identified by Dheeban, Deepak and Elis as being related with the following factors [9]:

- whether the covered concepts of the e-courses meet the expected learning target of the user, which depends on:
  - previous knowledge;
  - previous experience;
- whether the difficulty level of the e-learning material matches the ability level of the user, which depends on:
  - age;
  - level of education;
  - chosen learning subjects;
- the different learning time for users: a learner’s ability and attention influence the individual learning time;
- the weight of the learning concepts covered in an e-course, which has to be balanced;

E-learning platforms must not be just tools of content distribution of pedagogical resources and not taking into account the real interests of the learners: “…without personalization, e-learning is only ever going to be a generic mass produced experience and will tend towards a model of teaching that makes the computer a virtual lecturer, rather than a virtual personal tutor”[4].

In order to correct this drawback of e-learning, coming from the lack of personalization, researchers and educators try to use EAs algorithms, so that the e-education not to become “increasingly redundant and archaic”[4]. A first attempt in this direction consists in creating pedagogical paths, based on the learners’ profile and their learning objectives. Azough, Bellafkih and Bouyakhf used genetic algorithms to resolve the problem of searching the most optimal path from a starting point, represented by learners’ profile, to a final point, represented by learning objectives, while passing through intermediate points, represented by courses [2]. This optimal path is represented by an ordered list of courses. Each course requires pre-requisite concepts and offers a set of post-acquired concepts. The learner’s profile is built from all the concepts he/she has. The learning objective is represented by all the concepts he/she has to have. The coding used for representing the profile and the final goal is the binary coding. The genetic algorithm uses the same binary coding. Huang, Huang and Chen use genetic algorithms for curriculum sequencing [10], but they don't treat only the content problems, as Azough, Bellafkih and Bouyakhf do [2]: they also search for the most optimal teaching operation (presentation, example, question or problem). Huang, Huang and Chen argue that learners' ability should also be studied, when choosing the curriculum, besides considering learners' interests and browsing behaviors [10]. Other researchers use EAs for personalized recommendations in virtual learning environments [11], though supplying the human instructor. The hybridization of genetic algorithms, by combining them with fuzzy systems, is seen as a solution for intelligent tutorial applications [12]. Alexakos, Giotopoulos, Thermogianni, Beligiannis and Likothanassis combine genetic algorithms with Bayesian networks in order to provide intelligent assessment agents to an e-learning environment: Bayesian nets are used to model the sequence of the questions and the genetic algorithms are used to classify the users/ students into categories (according to their abilities) and to update the user model [13]. Other researchers use EAs instead of data mining, for rule discovery in learning management systems: they make rules which describe relationships between the students' usage of the different activities provided by an e-learning system and the final marks obtained at the courses. An interesting case study from Spain, using Moodle data, proves the practicability of using EAs in rule discovery [14].

Particle swarm optimization is a very useful EA related technique [15] [16] [17], with various variants [18]: 2-D Otsu PSO (TOPSO), Active Target PSO (APSO), Adaptive PSO (APSO), Adaptive Mutation PSO (AMPSO), Adaptive PSO Guided by Acceleration Information (AGPSO), Angle Modulated PSO (AMPSO), Attractive Repulsive Particle Swarm Optimization (ARPSO), Augmented Lagrangian PSO (ALPSO), Best Rotation PSO (BRPSO), Binary PSO (BPSO), Co-evolutionary PSO, Combinatorial PSO (CPSO), Comprehensive Learning PSO (CLPSO), Concurrent PSO (CONPSO), Constrained optimization via PSO (COPSO), Cooperative PSO (CPSO_M), Cooperative PSO (CPSO_S), Cooperatively Coevolving Particle Swarms (CCPSO), Cooperative Multiple PSO (CMPSO), Cultural based PSO (CBPSO), Dissipative PSO (DPSO), Divided range PSO (DRPSO), Dual Similar PSO Algorithm (DSPSOA), Dynamic adaptive dissipative PSO (ADPSO), Dynamic and Adjustable PSO (DAPSO), Dynamic Double Particle Swarm Optimizer (DDPSO), Dual Layered PSO (DLPSO), Dynamic neighborhood PSO (DNPSO), Estimation of Distribution PSO (EDPSO), Evolutionary Iteration PSO (EIPSO), Evolutionary Programming and PSO (EPPSO), Extended Particle Swarms (XPSO), Extended PSO (EPSO), Fitness-to-Distance Ratio PSO (FDRPSO) and so on and so forth. The diversity of the PSO variants conducts to a wide range of applications: Sedighizadeh and Masehian identified 41 such applications, in the field of electrical engineering, mathematics, chemical and civil engineering [18]. Sedighizadeh and Masehian didn't identify the e-education as a possible application domain for PSO, but other researchers suggest otherwise: Dheeban, Deepak, Dhamodharan and Elias show that PSO with inertia-coefficient is suitable for improving e-learning courses composition [9]. They also underline that their PSO variant is better than the Basic particle swarm optimization algorithm (BPSO) and experimental results come to strengthen their hypothesis. Ho, Yin, Hwang, Shyn and Yean use enhanced multi-objective PSO to improve the e-assessment services [19]: they try to solve the problems of multiple assessment criteria and parallel test sheets' composition from large item banks. Their proposed algorithm was compared to a competing genetic algorithm and they proved the superiority of PSO over classical genetic algorithm. For comparison purposes, they used performance metrics, metrics being a widely accepted practice in evaluation of information systems [20].

The current study shows another application of PSO to the e-assessment domain. The main purpose of the application is to enhance the formative features of e-assessment, which is not regarded only a knowledge evaluation instrument, but a knowledge creation one. In order to illustrate the suitability of PSO for resolving the formative e-assessment problem, the applied PSO algorithm is further described.

3 Particle Swarm Optimization solution to address formative e-assessment challenges in project management

3.1 Particle Swarm Optimization – algorithm description

PSO algorithm is a robust stochastic optimization technique, which is inspired from the movement and intelligence of swarms. PSO applies the concept of social interaction to problem solving. It was developed in 1995 by James Kennedy (social-psychologist) and Russell Eberhart (electrical engineer) [15]. It uses a number of particles that constitute a group moving around in the search space looking for the best solution. It imitates the bird from a flock which is nearest to the food. Each particle is treated as a point in a N-dimensional space which adjusts its “flying” according to its own flying experience as well as the flying experience of other particles. All particles have fitness values, evaluated through the fitness function and velocities. The two variables which are iteratively changed in PSO algorithm are the following ones:

- *pbest* (personal best): each particle keeps track of its coordinates in the solution space which are associated with the best solution (fitness) that has achieved so far by that particle;
- *gbest* (global best): another best value that is tracked by the PSO is the best value obtained so far by any particle in the neighborhood of that particle;

Each particle tries to modify its position using the following information: the current positions, the current velocities, the distance between the current position and *pbest*, the distance between the current position and *gbest*. Based on this idea, the PSO algorithm is illustrated in Fig.1: first, all considered particles are initialized with random positions and velocities. For all particles, the fitness function is evaluated and compared to the fitness value of the personal best position. If the current position has a better fitness value than the fitness (*pbest*), then current position becomes the new *pbest*. The best value of all *pbests* becomes the global best. The particles’ positions and velocities is updated according to this global best (*gbest*). The whole process happens until a certain number of iterations (*maxIter*) take place.

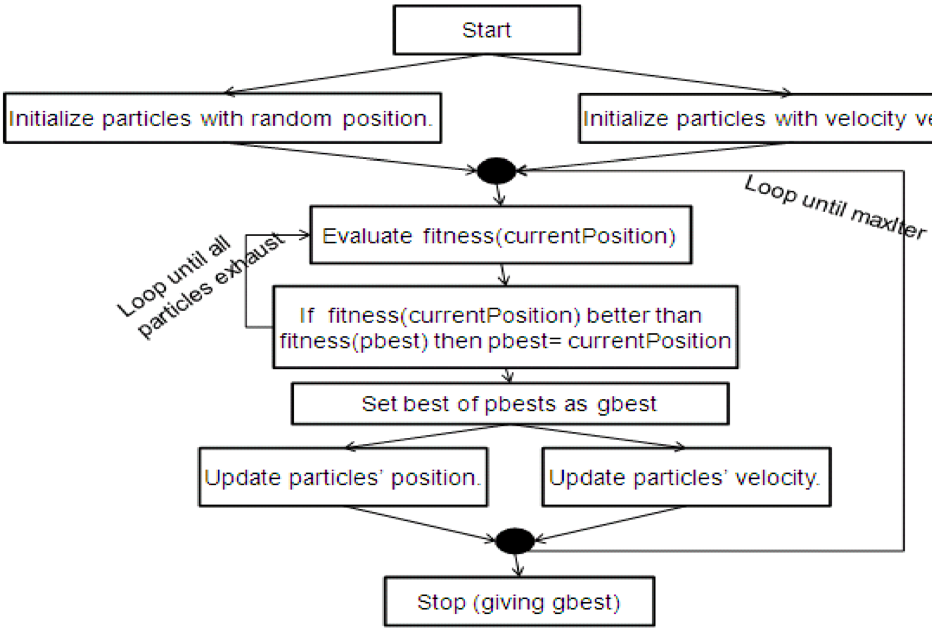

Fig. 1. PSO Algorithm

The final result of the algorithm is the *gbest* value. The fitness function has to be customized for each application of the PSO. The updates of the particles' position and velocity are made using the following formulas:

$$V_i^{k+1} = wV_i^k + c_1\, rand_1()\ x\ (pbest_i - s_i^k) + c_2\, rand_2()\ x\ (gbest - s_i^k) \quad (1)$$

$$s_i^{k+1} = s_i^k + V_i^{k+1} \quad (2)$$

where:

- v_i^k : velocity of particle *i* at iteration *k*
- w: weighting function (inertia weight)
- c_1, c_2 : learning factors
- $rand_1()$,$rand_2()$: uniformly distributed random number between 0 and 1
- s_i^k : current position of particle *i* at iteration *k*
- $pbest_i$: pbest of particle *i*
- gbest: gbest of the group

The first formula is used for updating the particles' velocity and the second formula is used for updating their positions.

Other useful information for using successfully a PSO approach is the following:

- a large inertia weight (*w*) facilitates a global search while a small inertia weight facilitates a local search; by linearly decreasing the inertia weight from a relatively large value to a small value one can obtain the best PSO performance compared with fixed inertia weight settings; the best solution for inertia factor is established by each implemented PSO solution;
- the two learning factors are constants with values between 0 and 4: their best values is established, usually, experimentally;
- the number of particles depends, again, on the PSO application; usually, the number varies between 20 and 40;
- the fitness function is strongly related to the representation of the problem to be solved by the PSO;

Before describing the PSO elements which are specific to the subject of this study, the problem addressed here by the PSO has to be clarified: the formative e-assessment.

3.2 Problem description for formative e-assessment in project management

Formative assessment is defined as "an ongoing process of monitoring learners' progresses of knowledge construction" [21] or "the process of seeking and interpreting evidence for use by learners and their teachers to decide where the learners are in their learning, where they need to go and how best to get there" [22]. Formative e-assessment is a much more complex process, because of the 'e' dimension, which lies over the assessment process itself. Its value was greatly appreciated by researchers: formative e-assessment promotes self-reflection, and students can take control of their own learning [23]. Formative e-assessment can fill a gap, between "the obtained (observed performance) and the intended (as defined by the objectives)"[24]. The author argues that it fills a knowledge gap, that's why formative e-assessment has an important role in learning and teaching, being known as "assessment for learning"[25]. Black and Wiliam propose a set of strategies to give e-assessment formative features [26]:

- (S1): engineering effective classroom discussion, questions, and learning tasks that elicit evidence of learning;
- (S2): providing feedback that moves learners forward;
- (S3): clarifying and sharing learning intentions and criteria for success;
- (S4): activating students as owners of their own learning;
- (S5): activating students as instructional resources for one another;

Wang completes them, by adding the following strategies [27]:

- (S6): repeat the test: correct answers aren't given, so that the test can be repeated till all the answers are correct;
- (S7): query scores: users are given other users' scores, so that they can compare themselves;
- (S8): ask questions: users can ask their tutors, using the e-mail;
- (S9): all pass and then reward: of the user answers correctly at the same question, three times in a row, then that question isn't given at other test sessions; otherwise, it is considered that the user guessed the answer, so the question remains in the questions' bank;

All the strategies mentioned underline the idea that a personalized e-assessment would increase its formative value. The author argues that a dynamic adaptive engine for questions generation in an e-assessment application would increases the formative value. When starting an e-test, the user must not receive a predefined set of questions: the user should get the questions suitable for his/her abilities and learning objectives. For project management (PM) domain, the selection of the most suitable next question depends on:

- ability level of the user, established by one of the following methods:
  - pre-test of 5 questions;
  - self-evaluation: a grade given by each user, based on his/her self-estimation of one's knowledge (previous courses in project management might help the user to accurately evaluate himself/ herself);
- desired difficulty of the e-test: each test can be used for preparing to certifications of level A, B, C or D [5], each of these certifications having another attached difficulty grade, as seen in Table 1;
- difficulty of the question: each question checks certain concepts from the International Project Management Association standard [28];
- past use of the question: according to strategy (S9) stated above, it's better for a user to answer correctly to a question three times in a row, but there is the possibility for that user to remember the correct answer from previous uses of the question; that's why the author considers that a trade-off between previous exposures of the question and previous answers given to that question should be made; the question with the lowest exposure number will be chosen, but this number will be weighed by a score regarding the correctness of previous answers; an example of this selection criteria is given in Table 2;
- importance of the selected question to the user's learning objectives; these learning objectives are reflected in the competences he/she wants to check; each project management competence has attached a set of concepts (an example is offered in Table 3) and each question checks a number of concepts: the more concepts are common, between the

targeted competence and the selected question, the more important is that question for the user's learning objectives;

Table 1. Difficulty grade attached to PM certification levels

Certification level	Difficulty grade
A	1
B	0.75
C	0.5
D	0.25

Table 2. Past uses of two questions and their influence to the questions' selection

Situation Pre-data	Situation Description	Situation Explanation
Qs1: the first question; Qs2: the 2nd question; ExpMax: the maximum exposure number of a question from the questions pool; ExpQs1: the exposure number for first question; ExpQs2: the exposure number for the 2nd question; CaQs1: number of times in which Qs1 has been correctly answered; CaQs2: number of times in which Qs2 has been correctly answered;	ExpMax = 10 ExpQs1 = 2 ExpQs2 = 3 CaQs1 = 2 CaQs2 = 1 Past use value for Qs1: $\frac{\text{ExpQs1} \times \text{CaQs1}}{ExpMax^2} = 0.04$ Past use value for Qs2: $\frac{\text{ExpQs2} \times \text{CaQs2}}{ExpMax^2} = 0.03$ The criteria applied for qs selection is the question having the minimum past use value. The selected question is Qs2.	Although Qs2 had a greater exposure number (3 instead of 2), it is preferred for the formative e-assessment, as it wasn't answered correctly as much as Qs1. Consequently, Qs2 is considered suitable for future test challenges.

Table 3. Concepts required by C1.19 - „Start-up" competence in the PM domain

Concept Code	Concept Description
DDI	Decision of making the investment
DIP	Document for initiating the project
PRO	Project proposal
CPR	Project charter
DDP	Decision to start the project
EEP	Pre-evaluation of the project
ATP	Assigning the project

3.3 Problem solution

Based on the PSO algorithm and on the assessment criteria described in the previous section, the elements necessary to resolve the formative e-assessment problem in project management are presented below. The updating of particles' velocities and positions is made using the formula 1 and formula 2.

- **Problem representation**

A = ability level of the user, where $0 \le A \le 1$

D = desired difficulty of the e-test, where $D \in \{0.25; 0.5; 0.75; 1\}$, as presented in Table 1

d_q = difficulty of the question *q*, where $0 \le d_q \le 1$

ExpMax = the maximum exposure number of a question from the questions pool

Exp_q = the exposure number for question *q*, where $0 \leq Exp_q \leq \text{ExpMax}$

CA_q = the number of times in which question *q* has been correctly answered

CO_q = number of concepts which are verified by the question q, contained by the competences established to be checked by the e-assessment session and unverified so far

$\text{Max}CO_q$ =number of concepts which are verified by the question q and contained by the competences established to be checked by the e-assessment session

- **Fitness function**

$$\text{Minimize } f(q) = \begin{cases} |d_q - A| + |d_q - D| + \frac{Exp_q \times CA_q}{ExpMax^2} + \frac{1}{CO_q}, if\ CO_q > 0 \\ |d_q - A| + |d_q - D| + \frac{Exp_q \times CA_q}{ExpMax^2} + \frac{1}{\text{Max}CO_q}, if\ CO_q \leq 0 \end{cases}$$

where *q* represents a question id

At each iteration, the algorithm searches for the question with the difficulty degree closest to the ability level set for the user (first term of the fitness function), the difficulty degree closest to the difficulty level set for the test, which wasn't correctly answered and not very exposed in previous test session (the 3rd term of the function) and last, but not least, which checks the biggest number of concepts from the learning objectives initially established (the 4th term of the function).

- **Inertia formula**

The inertia weight is calculated using the formula 3. The formula was used by other PSO researchers, also [16], [17], [18].

w = wIn-[(wIn - wF) x i]/maxIter (3)

where:

- wIn = initial weight
- wF = final weight
- i = current iteration number.
- maxIter = maximum iteration number

4 A formative e-assessment tool based on the proposed Particle Swarm Optimization solution

The PSO algorithm is included in the adaptation models which can be accessed by the adaptive engine of an e-assessment application for project management. For understanding the workflow of the application, the elements included in the application prototype (see Figure 2) are further described. The e-assessment application is developed using C# and ASP.NET technologies.

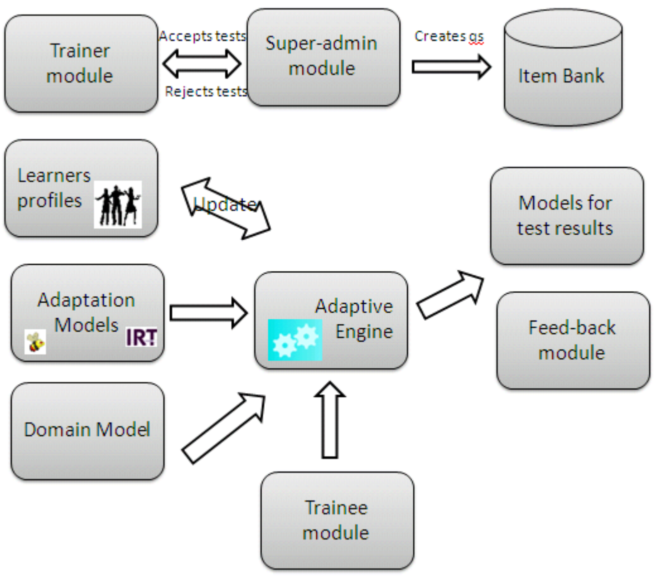

Fig. 2. Prototype of the e-assessment tool

Domain model

At conceptual level, the PM standard of the International Association of Project Management [28] is used. At programming level, *nHibernate* is exploited. A nHibernate mapping example for the object “question” (representing an item from a test) is:

```
<?xml version="1.0" encoding="utf-8" ?>
<hibernate-mapping
xmlns="urn:nhibernate-mapping-2.2"
assembly="CertExam.Domain.Model"
namespace="CertExam.Domain.Model">
  <class name="Questions, CertExam.Domain.Model"
table="Questions">

    <id name="Id" type="Guid" unsaved-value="null">
      <column name="Id" not-null="true" unique="true"/>
      <generator class="guid.comb" />
    </id>

    <property name="TextQs" type="String" column="TextQs" not-null="true" />
    <property name="TypeQs" type="Boolean" column="TypeQs" not-null="true" />
    <property name="IsUsableQuestion" type="Boolean" column="IsUsableQuestion" not null="false" />

    <many-to-one name="Category" class="Categories">
      <column name="Category" not-null="true"/>
    </many-to-one>

    <property name="Img" type="Byte[]" column="Img" length="1000000" not-null="false"/>

    <bag name="Answers" inverse="true" cascade="all">
      <key column="Question"/>
      <one-to-many class="Answers"/>
    </bag>

    <bag name="Concepts" inverse="true" cascade="all">
      <key column="Question"/>
      <one-to-many class="Concepts "/>
    </bag>

    <loader query-ref="GetImageOnId"/>
  </class>
```

A question has a category, a type (it can be an open question or a multiple-choice question), an image (if the case), a set of answers and a set of concepts.

User model

In order to establish the user ability level and learning objectives, the following elements are taken into account: an initial level of knowledge, the previous education in project management, objectives (what kind of project management certification the user wants – A, B, C or D certification), performance (previous scores), learning preferences (a check-list with what kind of PM competences are to be evaluated).

Adaptation models

There are different available adaption models: the PSO model, a model which uses the Item Response Theory and a rule-based model. The last model is defined by the test creator, through the trainer module. An example for this last model could be created on the following rule: “if (test D certification wanted) or (test C certification wanted) and (previous education includes PM) then (the 1st qs has 0.4 difficulty)”.

Models for test results

These models can pe predifined or customized by the test creator, through the trainer module.

Adaptive engine

The adaptive engine uses an adaptation model, updates the user profile, displays the results, offers feedback, accesses the Recommendation Engine, for indicating more bibliography to the user, though increasing the formative dimension of the e-assessment.

Modules of the e-assessment application

The four modules of the e-assessment application have the purpose of increasing the interactivity between the actors of an e-assessment system:

- Trainer module: creates the tests, chooses the proper questions, establishes the user profile, chooses an adaptive model, and sends invitations to users which have a profile suitable for a certain test.
- Super-admin module: creates the questions, accepts or rejects tests created by trainers.

- Trainee module: takes the tests, especially created for his profile or available in demo version (see Figure 3).
- Feed-back module: calls a Recommendation Engine and offers formative feed-back to users. After the test is finished, the user will receive his score and a list of questions correctly or incorrectly answered. The user will be allowed to return to any question and he/she will be able to find out what was wrong. This module permits the calling of a web crawler, which will provide links of useful documents, related to the incorrectly answered questions.

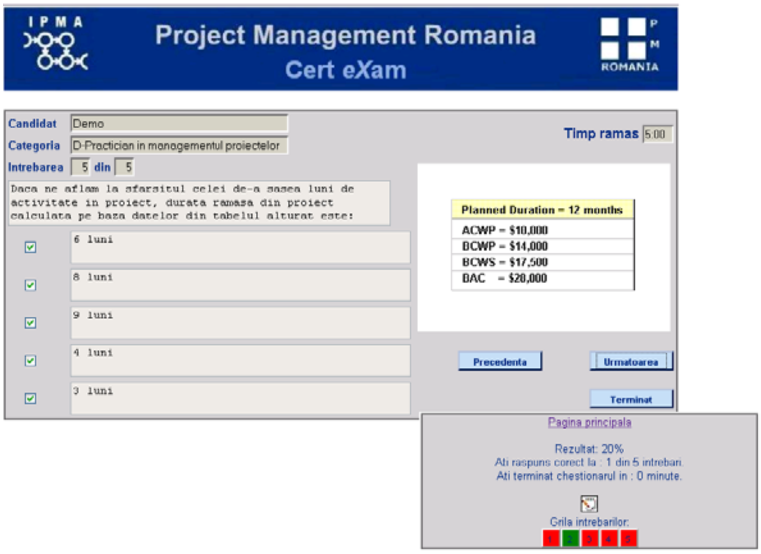

Fig. 3. Interface for the Trainee module in Project Management E-assessment

The workflow of accessing the PSO Algorithm, available in the adaptive modules component from Fig. 2 can be see in Figure 4.

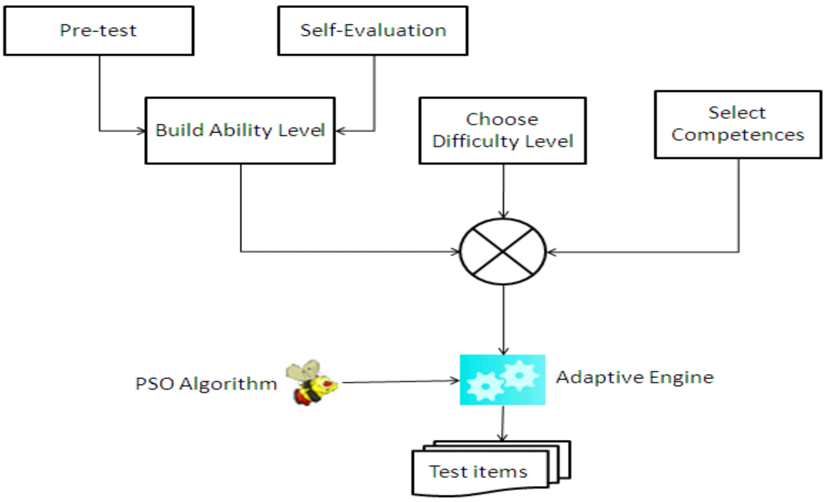

Fig. 4. Workflow of the PSO Scenario

5 Results and validation of the proposed Particle Swarm Optimization approach

Two attributes were checked for the proposed PSO algorithm, both of them in relation with the formative dimensions of the e-assessment tool: utility of the PSO e-tests and performance of PSO algorithm with respect to the other available adaptive algorithms in the system.

Method for utility validation

For evaluating the utility of the proposed algorithm, the following formula was used:

$$Utility_u = \frac{\sum_{i=1}^{n} P(u,i)}{n}, P(u,i)\epsilon[0,1] \quad (4)$$

Where:

$Utility_u$=the utility of the PSO algorithm seen by the user *u*

P(u,i) the utility of the PSO algorithm seen by use *u*, after session *i*

n= number of e-assessment sessions taken by the user

Method for performance validation

The performance of the PSO-algorithm was compared to IRT-algorithm (the algorithm using the Item Response Theory) and Rule-based algorithm. The method used for this was a questionnaire: the users were asked via e-mail to grade each of the 3 adaptation models, by using a grade from 1 to 10 and two points of view: performance regarding learning outcome and performance regarding feed-back times.

Experimental data

Five users were asked to participate at the experimental validation of the PSO e-assessment. All five users have graduated a project management master and wanted to obtain a level D certification in project management, according to International Project Management Association standards [28]. In Table 4, there are the utility grades awarded by the users.

Table 4. Utility Indicators for the PSO E-assessment

Users	Session1	Session2	Session3	Session4	Session5	Utility
User1	0.6	0.7	0.7	0.8	0.88	0.736
User2	0.5	0.5	0.4	0.4	0.5	0.46
User3	0.8	0.81	0.82	0.85	0.9	0.836
User4	0.7	0.75	0.75	0.8	0.88	0.776
User5	0.66	0.67	0.78	0.78	0.9	0.758

The utility trend is ascendant for all the users, except User2, which had a pretty low ability level. The utility indicators are quite satisfactory. For the experimental data to be more accurate, more users should be asked to evaluate the tool.

According to the Figure 5, the five users consider PSO-algorithm to be more efficient with respect to the achieving of learning objectives than the IRT-algorithm, but less efficient than the rule-based one. Probably, if the item bank would be increased (now, only 200 questions were used), the performance of PSO algorithm will also increase. When analyzing the feed-back response, the situation is similar.

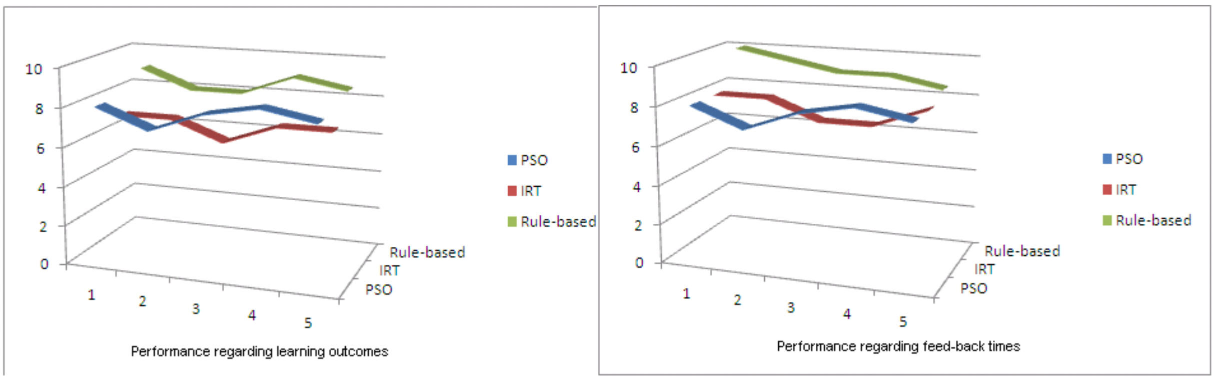

Fig. 5. Performance of the Adaptation Algorithms in the A-Assessment Tool

6 Conclusions

The current study highlights the benefits of using an evolutionary algorithm to the e-education domain, in general and to e-assessment in project management, especially. The proposed Particle Swarm

Optimization method offers formative value to e-assessment and transforms it into a learning tool. Formative assessment has been studied in various educational systems, from countries like Australia (Queensland), Canada, Denmark, England, Finland, Italy, New Zealand and Scotland, and turned out to be highly efficient "in raising the level of student attainment, increasing equity of student outcomes, and improving students' ability to learn" [29]. The world-wide interest for the formative e-assessment makes the study valuable and interesting to the current research directions. As future improvements, the fitness function of the PSO approach can be changed, by adding a more accurate description for the fourth term, in which the correlation between the concepts checked by the item tests and the ones established in the learning objectives is depicted. Further experiments are needed, with different learning coefficients for the PSO algorithm and more questions available in the items' pool.

Acknowledgement
This article is a result of the project „Doctoral Program and PhD Students in the education research and innovation triangle". This project is co funded by European Social Fund through The Sectorial Operational Program for Human Resources Development 2007- 2013, coordinated by The Bucharest Academy of Economic Studies.

References

[1] T. Blickle, "Theory of Evolutionary Algorithms and Application to System Synthesis," Swiss Federal Institure of Technology , Zurich, PhD Thesis 11894, 1996.

[2] S. Azough, M. Bellafkih, and E. H. Bouyakhf, "Adaptive E-learning using Genetic Algorithms," *IJCSNS International Journal of Computer Science and Network Security*, Vol. 10, no. 7, pp. 237-277, 2010.

[3] C. M. Mark, "Genetic Algorithms," Annals of Tibiscus University, *Computer Science Science Series*, Vol. 1, No. 1, pp. 72 - 89 , May 2004.

[4] H. Ashman, T. Brailsford, and P. Brusilovsky, "Personal Services: Debating the Wisdom of Personalisation," in *Lecture Notes in Computer Science (5686) - Advances in Web based Learning*, Aachen, 2009, pp. 1-11.

[5] C. Bodea and M. Dascalu, "A Parameterized Web-based Testing Model for Project Management," in *Lecture Notes in Computer Science (5686) - Advances in Web based Learning*, Aachen, 2009, pp. 68-72.

[6] C. Bodea, M. Dascalu, and M. Coman, "Quality of Project Management Education and Training Programmes," in *Communications in Computer and Information Science (73) - Technology Enhaced Learning*, Quality of Teaching and Educational Reform, Athens, 2010, pp. 324-330.

[7] M. Dascalu, C. Delcea, D. Palaghita, and B. Vintila, "The Impact of Competences Assessment Systems to Enterprise Performance A Study on Project Management E-Assessment," in *Proceedings of 12th International Conference on Enterprise Information Systems*, Funchal, Madeira, 2010, pp. 59-64.

[8] C. Delcea, M. Dascalu, L. Dascalu, L. Lica, and M. Coman, "Evaluating the E-learning Impact on Company's Non-Financial Performance," in *EDULEARN10 Proceedings*, Barcelona, 2010, pp. 5714-5719.

[9] S. G. Dheeban, V. Deepak, L. Dhamodharan, and Susan Elias, "Improved personalized e-course Composition Approach using Modified Particle Swarm Optimization with Inertia-Coefficient," International Journal of Computer Applications, Vol. 1, No. 6, pp. 102-107, 2010.

[10] M-J. Huang, H-S. Huang, and M-Y. Chen, "Constructing a personalized e-learning system based on genetic algorithm and case-based reasoning

approach," *Expert Systems with Applications*, Vol. 33, pp. 551–564, 2007.
[11] A. Baylari and Gh. A. Montazer, "Design a personalized e-learning system based on item response theory and artificial neural network approach," *Expert Systems with Application*, Vol. 36, pp. 8013–8021, 2009.
[12] I. A. Uţă, R. Mihalca, A. Andreescu, I. Întorsureanu, and Ş. Kovacs, "The Possibility to Use Genetic Algorithms and Fuzzy Systems in the Development of Tutorial Systems," *Informatica Economică Journal*, Vol. 40, No. 4, pp. 34-38, 2006.
[13] C.E. Alexakos, C.K. Giotopoulos, E.J. Thermogianni, G.N. Beligiannis, and S.D. L. Likothanassis, "Integrating E-learning Environments with Computational Intelligence Assessment Agents," *World Academy of Science, Engineering and Technology*, Vol. 19, pp. 117-122, 2006.
[14] C. Romero, P. Gonzalez, S. Ventura, M.J. Del Jesus, and F. Herrera, "Evolutionary algorithms for subgroup discovery in e-learning:A practical application using Moodle data," *Expert Systems with Applications*, Vol. 36, pp. 1632–1644, 2009.
[15] M.A. Montes de Oca. (2007, May) Institut de Recherches Interdisciplinaires et de Developpements en Intelligence Artificielle. [Online]. http://iridia.ulb.ac.be/~mmontes/slidesCIL/slides.pdf
[16] J.F. Schutte and A.A. Groenwold, "A Study of Global Optimization Using Particle Swarms," *Journal of Global Optimization*, Vol. 31, no. 1, pp. 93 - 108, January 2005.
[17] M.E.H. Pedersen and A.J. Chipper, "Simplifying Particle Swarm Optimization," *Applied Soft Computing*, Vol. 10, No. 2, pp. 618-628, March 2010.
[18] D. Sedighizadeh and E. Masehian, "Particle Swarm Optimization:Methods, Taxonomy and Applications," *International Journal of Computer Theory and Engineering*, Vol. 1, No. 5, pp. 1793-8201, December 2009.
[19] T-F. Ho, P-Y. Yin, G-J. Hwang, S.J. Shyu, and Y-N. Yean, "Multi-Objective Parallel Test-Sheet Composition Using Enhanced Particle Swarm Optimization," *Educational Technology & Society*, Vol. 12, No. 4, pp. 193–206, 2009.
[20] I. Ivan and C. Ciurea, "Validation of Metrics for Collaborative Systems," *Informatica Economică Journal*, Vol. 48, No. 4, pp. 83-86, 2008.
[21] J-L. Hsu, H-W. Chou, and H-H. Chang, "EduMiner: Using text mining for automatic formative assessment," *Expert Systems with Applications*, 2010.
[22] Assessment Reform Group (ARG), "Assessment for learning: 10 principles," Cambridge, UK, 2002.
[23] N. Pachler, C. Daly, Y. Mor, and H. Mellar, "Formative e-assessment: Practitioner cases," *Computers and Education*, vol. 54, pp. 715–721, 2010.
[24] D.R. Sadler, "Formative assessment and the design of instructional systems," *Instructional Science*, Vol. 18, pp. 118–144, 1989.
[25] M. Birenbaum, H. Kimron, H. Shilton, and R. Shahaf-Barzilay, "Cycles of inquiry: Formative assessment in service of learning in classrooms and in school-based professional communities," *Studies in Educational Evaluation*, vol. 35, pp. 130–149, 2009.
[26] P. Black and D. Wiliam, "Developing the theory of formative assessment," *Evaluation and Accountability*, Vol. 21, No. 1, pp. 5–31, 2009.
[27] T.-H. Wang, "Web-based quiz-game-like formative assessment: Development and evaluation," *Computers and Education*, Vol. 51, pp. 1247–1263, 2008.
[28] International Project Management Association. (2006, June) International Project Management Association Competence Baseline. [Online]. http://www.pm.org.ro/ICB-V-3-0/ICB-V-3-0.pdf

[29] OECD. (2005) OECD. [Online]. www.oecd.org/dataoecd/19/31/35661078.pdf

Maria-Iuliana DASCĂLU has a Master Degree in Project Management from the Academy of Economic Studies, Bucharest, Romania (2008) and a Bachelor Degree in Computer Science from the Alexandru-Ioan Cuza University, Iasi, Romania (2006). She is a PhD student in Economic Informatics at the Academy of Economic Studies, combining her work experience as a programmer with numerous research activities. Her research relates to computer-assisted testing with applications in e-learning environments for project management, competences development systems and their benefits to lifelong learning. Maria Dascălu is a Certified Project Management Associate (2008). She also conducted a research stage at the University of Gothenburg, Sweden, from October 2009 to May 2010. She published, as a sole author or in collaboration, 3 ISI articles, 7 papers in ISI Proceedings, 10 articles in journals indexed in international databases and she presented more than 15 papers at national and international conferences.